

2024/5/22

刘梓桐

第一章 操作系统引论

一、名词解释

1. 操作系统: 集成软件和硬件资源, 管理软石文件的系统
4. 实时操作系统: 严格时限内响应事件的操作系统
5. 互斥共享: 确保同一时间仅一个进程访问临界资源

三、填空题

1. 一个完整的计算机系统包括 (硬件系统) 和 (软件系统) 两部分
3. 计算机系统中的资源包括 (硬件资源) 和 (软件资源) 两部分
5. 操作系统的特征包括并发性、(共享性)、(异步性) 和虚拟性

第二章 进程的描述与控制

一、名词解释

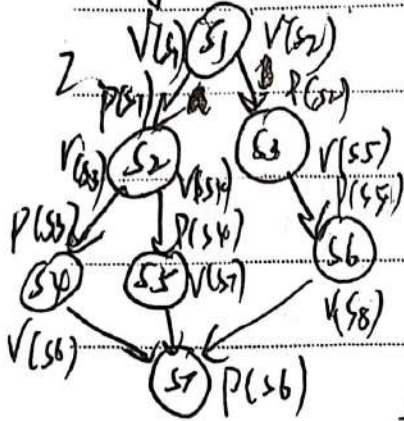
3. 临界区: 进程中访问共享资源的代码段, 需互斥执行。
4. 进程同步: 协调并发进程的执行顺序以避免数据冲突。

三、填空题

1. 进程特性包括其结构特征、(动态性)、并发性 (异步性)、独立性
和结构性。
5. 在比较进程和程序时, 进程是 (动态) 的概念, 而程序是 (静态) 的概念。
6. 因共享公有资源而造成对并发程序执行速度的制约称为 (间接制约)。
8. 进程通信包括 (共享内存)、(消息传递) 和 (管道通信) 三种方式。
9. 线程的基本状态包括 (就绪) 状态、(运行) 状态 和 (阻塞) 状态。



五、简答题



$S_1 \rightarrow S_2, S_3$

$S_2 \rightarrow S_4, S_5$

$S_3 \rightarrow S_6$

$S_4, S_5, S_6 \rightarrow S_7$

$i: S_1 \rightarrow S_2; b: S_1 \rightarrow S_3; c: S_2 \rightarrow S_4; d: S_2 \rightarrow S_5; e: S_3 \rightarrow S_6$

$f: S_4 \rightarrow S_7; g: S_5 \rightarrow S_7; h: S_6 \rightarrow S_7$

Semaphore $a, b, c, d, e, f, g, h;$

$a.value = b.value = c.value = d.value =$

$e.value = f.value = g.value = h.value =$

0;

begin

{ $s_1; \text{signal}(a); \text{signal}(b);$ wait(a);

wait(a); $s_2; \text{signal}(c); \text{signal}(d);$ }

{wait(b); $s_3; \text{signal}(e);$ }

{wait(c); $s_4; \text{signal}(f);$ }

{wait(d); $s_5; \text{signal}(g);$ }

{wait(e); wait(f); wait(g); wait(h); $s_7; \text{signal}(h);$ }

{wait(h); wait(g); wait(h); s_7 }

end

}

2024010002

54343

3.

Semaphore mutex = 1;

empty = 1; apple = 0; orange = 0;

Father()

{ while (1)

{ ~~P(empty);~~ ~~wait(empty);~~ P(empty);

if (apple > 0) V(s1);

else V(s2);

}

}

void Son1()

{ while (1)

{ P(orange);

P(mutex);

~~wait(mutex);~~

V(empty);

V(mutex);

~~wait(mutex);~~

}

}

void Daughter()

{ while (1)

{ P(apple);

P(mutex);

~~wait(mutex);~~

V(empty);

V(mutex);

~~wait(mutex);~~

}

}

4. def P1():

while True:

P(empty1)

read-from-disk-to-buffer1()

V(full1)

def P2():

while True:

P(full1)

P(empty2)

copy-buffer1-to-buffer2()

V(empty1)

V(full2)

Maxleaf



扫描全能王 创建

```
def p1():
```

```
    while True:
```

```
        p(full2)
```

```
        print butter21)
```

```
        v(empty2)
```

```
def barber():
```

```
    while True:
```

```
        p(customers)
```

```
        p(mutex)
```

```
        serve_customer()
```

```
        v(sofa)
```

```
        v(mutex)
```

```
        cut_hair()
```

```
        p(payment)
```

```
        v(hair_barber)
```

```
def customer():
```

```
    p(mutex)
```

```
    if sofa == 0:
```

```
        v(mutex)
```

```
        leave()
```

```
    else:
```

```
        p(sofa)
```

```
        v(customers)
```

```
        v(mutex)
```

```
        p(barber)
```

```
        get_haircut()
```

```
        pay()
```

```
        v(payment)
```

操作系统调度与死锁

1. 名词解释: 从就绪队列中选择进程并分配CPU资源的过程

2. 处理器调度: 为进程分配CPU的决策过程

3. 周转时间: 从作业提交到完成所需的全部时间

4. 死锁: 进程互相等待资源无法推进的阻塞状态。

三要素:

2. 作业在其生存期间会经历(提交)、(就绪)、(执行/完成)状态

(死锁)的处理方法包括(预防)、(避免)、(检测)和(解除)

五. 问答

1. ① FCFS 调度: A B C D E

	到达	服务	TAT	WT	带权TAT
A	0	3	3	0	1
B	2	9	7	1	1.17
C	4	13	9	5	2.25
D	6	18	12	7	2.4
E	8	20	12	10	6

平均周转: $\frac{3+7+9+12+12}{5} = 8.6$
 平均带权周转: $\frac{1+1.17+2.25+2.4+6}{5} = 2.56$

② 非抢占 SJF: A B C D E

	到达	服务	TAT	WT	带权TAT
A	0	3	3	0	1
B	2	6	9	1	1.17
C	4	4	9	5	2.25
E	8	2	13	5	3.5
D	6	5	15	9	2.8

平均周转: $\frac{3+9+9+13+15}{5} = 8.6$
 平均带权周转: $\frac{1+1.17+2.25+3.5+2.8}{5} = 2.56$

③ 抢占 SJF: A B C D E

	到达	服务	TAT	WT	带权TAT
A	0	3	3	0	1
B	2	6	15	7	2.17
C	4	4	8	0	0
D	6	5	14	9	2.8
E	8	2	10	0	0

平均周转: $\frac{3+15+8+14+10}{5} = 8.2$
 平均带权周转: $\frac{1+2.17+0+2.8+0}{5} = 1.594$

④ HRRN: A B C D E

	到达	服务	TAT	WT	带权TAT
A	0	3	3	0	1
B	2	6	7	1	1.17
C	4	4	9	5	2.25
D	6	2	14	9	2.8

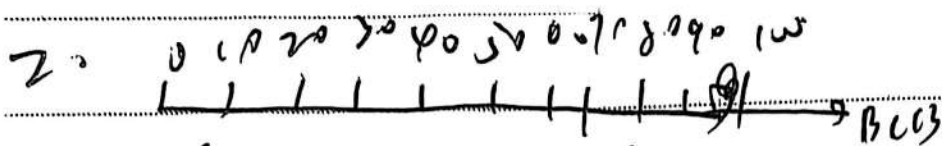
平均周转: $\frac{3+7+9+14}{4} = 8$
 平均带权周转: $\frac{1+1.17+2.25+2.8}{4} = 2.144$

⑤ RR (时间片轮转): A B C D E

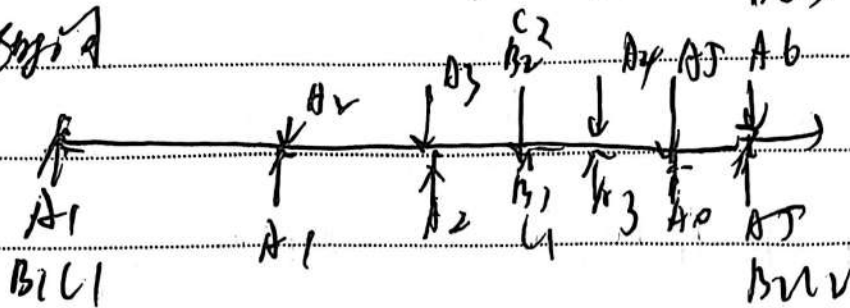
	到达	服务	TAT	WT	带权TAT
A	0	3	3	0	1
B	2	6	19	11	2.83
C	4	4	17	7	2.73
D	6	5	16	9	2.8
E	8	2	5	3	2.5

平均周转: $\frac{3+17+11+16+5}{5} = 10.8$
 平均带权周转: $\frac{1+2.83+2.73+2.8+2.5}{5} = 2.71$

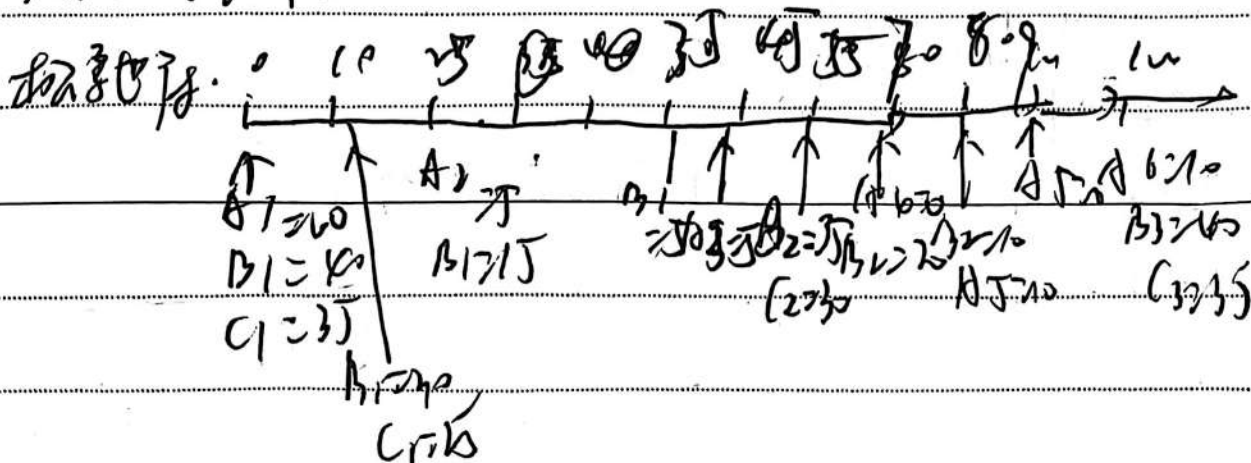




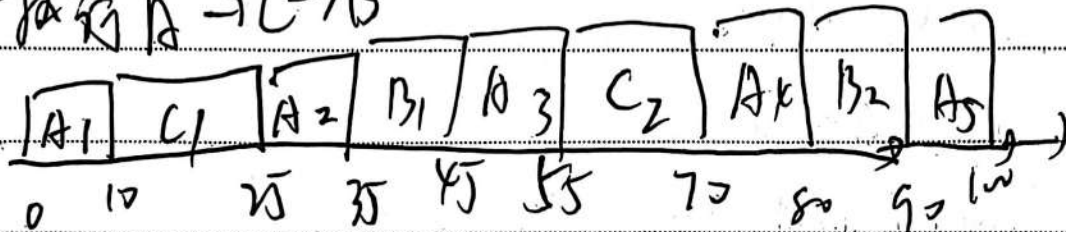
到达时间



到达时间



3. 调度规则 A → C → B



3. (1) Allocation

Process Allocation

P0 0 0 3 2

P1 1 0 0 0

P2 1 3 5 0

P3 0 3 3 2

P4 0 0 1 0

Maxleaf



扫描全能王 创建

安全检测过程:

进程	资源情况	work				need				Allocation				Finish
		A	B	C	D	A	B	C	D	A	B	C	D	
P0		1	6	2	2	0	0	1	2	0	0	3	2	True
P3		1	6	5	4	0	6	5	2	0	3	2		True
P0		1	9	8	6	6	6	5	6	0	0	1	X	True
P1		1	9	9	10	1	7	5	0	1	0	0	0	True
P0		2	9	9	10	2	3	5	6	1	3	5	X	True

可执行进程到 $\{P_0, P_3, P_0, P_1, P_1\}$: 系统安全

② $Request_2(1, 2, 2, 2) \leq Need_2(2, 3, 5, 6)$

③ $Request_2(1, 2, 2, 2) \leq Available(1, 6, 2, 2)$

④ 将 P_2 所需资源分配如下:

$Available(0, 4, 0, 0)$ $Allocation_2(2, 5, 7, 8)$

$Need_2(1, 1, 3, 4)$

⑤ 安全检查: 当前所有进程都不满足 $Need_i \leq Available$

$(0, 4, 0, 0)$: 系统不安全: 系统不能将资源分配给 P_2 的请求

(3) 系统有 3 个进程进入阻塞状态, 因为此时上述 3 个进程所需

的资源, 因得不到资源进入阻塞状态, 只有上述 3 个进程

能执行时, 系统才能运行, 因得不到资源阻塞

形成了循环, 此时系统会进入死锁状态

